INSE 6260

Software Quality Assurance

Acceptance Test Plan

Name	**Student Id**
Nidhi Arora	40014504
Karun Sharma	40016964

Date: (5/12/2017)

Software Quality Assurance

MiniSIS

APPROVAL CLIENT

Client:	Concordia University	Signature
Division:	Administration Division	
Department	Technical Department	
Function	Project Manager	
Location	1455 De maisonneuve Boulevard W. Montreal, Quebec, Canada H3G 1M8.	
Telephone	514-848-2424	
E-Mail address	rachida.dssouli@concordia.ca	Date (mm/dd/yyyy): 12/5/2017

MANAGEMENT SUMMARY

Project objective

The main objective of this project is to provide students and admins with an online application through which they can manage their courses. The users will be able to register for courses, pay their fees and get their transcript.

Test objective and assignment

The Objective of Acceptance test plan is to check the correctness of system. The main objective is to check whether all the requirements that are mentioned in System Requirement Specification document are satisfied or not. Also, It is used to find the faults and defects in the system if any.

Short description of the test approach

Testing will be undertaken by quality assurance team on the Mini Student Information System(MiniSIS) to check the functionality of system. Also, QA team will check the performance of system and ensure that all the functional and nonfunctional requirements conform to the SRS. QA team will run the predefined test cases on system and based on that report will be generated which provides the detail of passed and failed cases.

Qualitative objectives

Acceptance test phase should be completed on time and meet the acceptance criteria.
All the functional requirements and nonfunctional requirements stated in the SRS document need to check if they are working properly or not.

Go/no-go decisions

Once testing for system is done after execution of all the test cases by the QA team, the results are presented in form of a report and the project manager analyzes the results before accepting or neglecting the software.

Software Quality Assurance
MiniSIS

TABLE OF CONTENTS

1 INTRODUCTION

1.1 Goal of the Test Plan

The goal of this Test Plan (TP) for Acceptance Testing is to inform about the activities that are involved in the testing process, the test process involved in it and the deliverables concerning acceptance test plan for Mini Student Information System.

1.2 Assignment

1.2.1 Client

Here Concordia University is a client who has requested an acceptance test plan for the testing phase. This Acceptance plan will be used by the QA team for the Concordia University to run tests on the behalf of Concordia University's customer.

1.2.2 Supplier

Quality assurance team for Concordia University supplies the acceptance test plan and is responsible for executing all test cases and checking the outputs.

1.2.3 Assignment

This acceptance test plan acts as a guideline for QA team to create a report of all the results of test cases executed on the system. All observed and expected outputs are enlisted in this document. It is used to verify if the system presented follows or abides by all the requirements enlisted in the SRS document.

1.2.4 Scope

The system provides the services based on different classes of users. Those different classes will have different access levels to the system and its features. System administrator will have set of features to manage and maintain accounts of other users, while a faculty member will have access to the faculty calendar management, and course management area. Student account will let the user register or drop courses based on the defined rules of the faculty, see grades and transcripts, and pay tuition fees.
The system to be implemented will not support transferring students from a different school or carrying credits of transferred students between different departments.

Within scope:

The acceptance test plan would test all the basic enlisted in the SRS for the Student Information System. Checks on all the functional and nonfunctional requirements as discussed in section 3.1 are in the scope of this document.

Out of scope:

Unit testing, module testing and integration testing are not within the scope of this document that is responsibly of development team. Also, due to insufficient time to check, non-functional requirements as Performance, Maintainability, Portability cannot be checked.

1.2.5. Preconditions and assumptions

The following demands apply to the test process:

- Acceptance test plan in a part of total quality approach to Student Information System as mentioned in project plan.

To make the test process successful the following things need to be arranged:

- The developers should be well acquainted with the databases used by the University.
- Concordia University should clearly mention their expectations form the software.
- The testing strategy used in the acceptance testing should be in accordance to Concordia University's demands.
- The Project manager should make sure that the development team gets the required access to the database. Also, all the technical documents required for the project are made available.
- Unit testing and Integration testing are taken care by the development team and the report is provided to the QA team.

1.2.5 Acceptants and acceptance criteria

Acceptants

Acceptants on behalf of the commissioning organization are:

Name	Function	Department
Dr. Rachida Dssouli	Project manager at Concordia University	Academic Division

Acceptance criteria

The acceptance criteria for system and acceptance tests are:

Description	Norm
It is expected that all the test cases pass with a certain output range and criteria mentioned in section 4.4. All the observed outputs are same as expected outputs	Observed output = Expected output
In case where test cases are expected to fail then suitable error messages or outputs are displayed as mentioned in section 4.4 observed output or error message is same as expected output or error message	Observed output or error message = Expected output or error message
Proper error messages should be displayed in case when	Test cases should test

an un-expected input is encountered by the system.	system on un expected inputs.
All the non-functional requirements mentioned in SRS document should by satisfid by the test cases. Non-Functional requirements such as performance, reliability, correctness, and other standards that were agreed to in SRS.	All the non-functional requirements that are mentioned in SRS of Concordia University must pass the test cases
Defects mentioned in test report must be fixed by development team.	The resolved defects should be verified by the QA Team

Moment of release

The production phase of the project depends on the test report which is generated by executing the test plan as mentioned in acceptance test document. In case all requirements are met, the product is released.

2 DOCUMENTATION

This chapter describes the documentation used in relation to the development tests, often the same as the development basis. The documentation that has a relation with the master test plan will also be described in detail.

2.1 Basis for the test plan

The following documents were used as a basis for this test plan.

Document name	**Version**	**Date(mm/dd/yyyy)**	**Author**
Software Requirement Specification for MiniSIS	1.0	09/26/2017	Development Team

2.2 Test basis

The test basis consists of the documents from which the test cases are being derived. It contains the documentation that serves as a basis for the tests to be executed. The overview below states the test basis for the system and acceptance tests.

Document name	**Version**	**Date**	**Author**
Software Requirement Specification for MiniSIS	1.0	09/26/2017	Development Team

3 TEST STRATEGY

Taking into consideration the time constraint everything cannot be tested thoroughly. Therefore we have decide the task priorities for the test cases. These task priorities will decide the test capacity as effective as possible. The Test Plan document for MiniSIS consists the details of different task priorities in the test strategy section.

The test strategy determines how, when and with what thoroughness is the task should be tested. The motive of test strategy is to find the most critical defects in early stage of process al the least cost.

Execution of risk analysis is the first step in determining the test strategy in the Test Plan. The results of that Product Risk Analyses (PRA) that are relevant for the Acceptance test plan can be found in appendix 1. In this test plan the test strategy from the master test plan has been further elaborated for Acceptance test plan.

The test cases are enlisted in a document that accompanies this document. The traceability of test cases to the characteristics of software mentioned in SRS are given in the Table below.

3.1 Test strategy User Acceptance Testing

Characteristic/object part	PRA-RC	<Test level>	Test type	Test Case
Functionality				
Request Confirmation Email	C	●	FAT, UAT	TC002
Request Password Reset	C	●	FAT, UAT	TC003
Mobile Phone Number Confirmation	C	●	FAT, UAT	TC004
Two-Factor Authentication	B	●●●	FAT, UAT	TC001
Complete Account Registration	A	●●●	FAT, UAT	TC001
Edit User Profile	A	●●●	FAT, UAT	TC006
Log Actions	C	●	FAT, UAT	TC005
Create New User(System Admin)	A	●●●	FAT, UAT	TC001
Enable/Disable User Account(System Admin)	A	●●●	FAT, UAT	TC001
Register Course	A	●●●	FAT, UAT	TC007
Drop Course	A	●●●	FAT, UAT	TC008
View Transcript	A	●●●	FAT, UAT	TC0011
Pay Tuition Fees	A	●●●	FAT, UAT	TC0012
Manage Courses(Faculty Member,Program Advisor)	A	●●●	FAT, UAT	TC007, TC008
Manage Course(Instructor)	A	●●●	FAT, UAT	TC007, TC008
Add Course Grades	A	●●●	FAT, UAT	TC013
Total	A	●●●	System test	
Performance		I	User Acceptance	TC015
Security		I	User Ac-	TC014

			ceptance	
Reliability		I ●●	FAT, UAT	TC014
Note*: For detail of test cases ,please refer document : Acceptance Test Cases				

Note:- Here it is assumed that the developer has done the unit and the Unit Integration testing

<blank>	If a cell is blank, it means that the relevant test or evaluation level does not have to be concerned with the characteristic

Explanation for the table above:

PRA-RC	Risk class (from product risk analysis, where A=high risk, B=average risk, C=low risk)
Evaluation	Evaluation/review of the various intermediary products (requirements, functional design, technical design)
Development test	Unit test and Unit integration test
ST	System test
FAT	Functional acceptance test
UAT	User acceptance test
Impl	Implementation
●	Limited thoroughness of the dynamic test
●●	Medium thoroughness of the dynamic test
●●●	High thoroughness of the dynamic test
S	Static testing (checking and examining the products without executing the software)
I	Implicit testing (including in another test type without creating specifically designed test cases)

4 APPROACH

This section describes *how* the testing is handled in conformity with the test strategy. The test design table in section 4.1 gives an overview of this approach. A more extensive description for each part of the test approach will be given in section 4.2.

4.1 Testing Levels for MiniSIS

Test level	Goal	Responsibility
Unit Testing	For testing a unit of code while development process is going on.	Development Team
Integration Testing	For testing integration of various modules after they are integrated and made into software.	Development Team
System Test	For testing the system for all functional and non-functional requirements that are agreed to in SRS document.	QA Team
Functional Testing	Reviewing the functionality of the System for end users	QA Team
Acceptance test plan	To test for errors in functionality and usability errors	QA Team

4.2 Description test approach of acceptance test plan

4.2.1 Intake test object

The acceptance test plan starts with the execution of an intake of the test object in order to verify that the entry criteria are met. This intake consists of a completeness check and a pre-test.

Completeness check
The best way to ensure completeness check is to create a checklist of all the test objectives with a list of all the required documents.

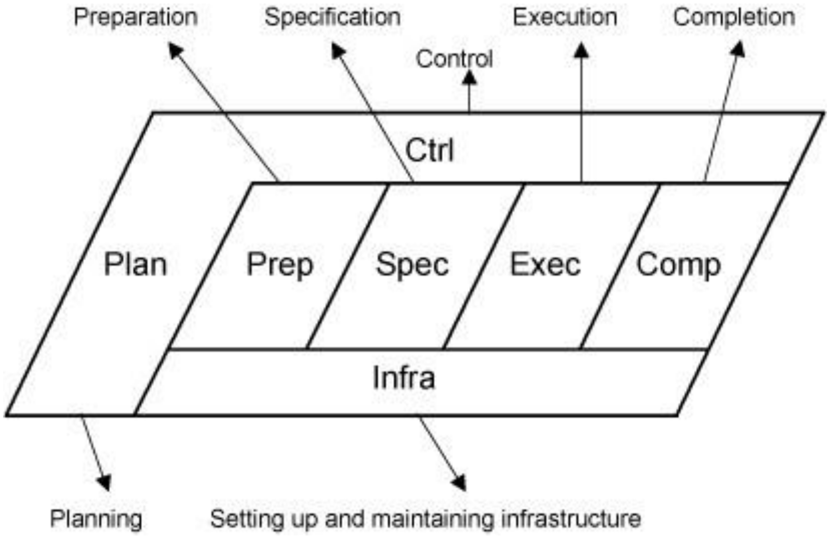
Pre-test
After installation of the test object, a pre-test takes place in order to determine whether the test object is good enough to start testing. The pre-test will be executed as follows:

1. Check whether all the basic functionality of MiniSIS that is adding courses, deleting courses, viewing transcript, paying online etc.
2. Secondly, for all the above functionalities, test cases should be developed to test all the possible scenarios that is success scenario and alternate or failure scenario. These also be tested on all the boundary values.
3. Finally, the design for test cases should be done keeping in mind the functionality and usability from end users point of view.

4.2.2 Acceptance Test Plan

- Week 1: Using a black box approach the system administrator module will be tested for defects such as links that do not work and missing links. All functions will be tested with correct and incorrect input.
- Week 2: The same procedure as week 1 but using a white box approach. Input will be given to all functions with the intention of calling internal functions.
- Week 3: Static testing. The code will be verified to look for uninitialized variables and other invalid code.
- Week 4: Verification that the module meets the system requirements specification and the system design specification.

4.3 Phasing Acceptance Testing



In the **Planning** phase, the test manager provides a plan that is supported by the client to adequately execute the test assignment. This is laid down in the test plan. In the **Control** phase the activities in the test plan are executed, monitored, and adjusted if necessary. The **Setting up and maintaining infrastructure** phase aims to provide the required test infrastructure that is used in the various phases and activities. The **Preparation** phase aims to have access to a test basis, The SRS document, agreed to with Concordia University who is client, should make sure that we have test cases of adequate quality and expectations for the MiniSIS.

The tests are specified in the **Specification** phase and executed in the **Execution** phase. This provides insight in the quality of the test object. The test assignment is concluded in the **Completion** phase. This phase offers the opportunity to learn lessons from experiences gained in the project. Furthermore activities are executed to guarantee reuse of products [1].

4.4 Entry and exit criteria

The following entrance criteria are defined for the phase Specification and Execution:

Entrance criteria for the Specification phase:

- All the test basis documents that are used for creating acceptance test plan have been shown and approved by Concordia University before creating test cases.
- All initial test phases namely unit, Integration and module testing is carried out properly and all defects that were found are fixed. All the documentation that is reports that were generated for these tests is submitted.
- Final test plan should be approved by Concordia University.

Entrance criteria for the Execution phase:

- All initial test phases namely unit, Integration and module testing is carried out properly and all defects that were found are fixed. All the documentation that is reports that were generated for these tests is submitted
- All the test scripts for acceptance test plan are submitted and are approved by Concordia University.
- The environment for testing is set in accordance with requirements for infrastructure, functionality and test data sets.
- There are no defects that are pending from initial testing phases unit module and integration that may hamper acceptance test plan. These defects will have such an impact on MiniSIS that no operational line, technical processes, administrative processes can be tested.

The following exit criteria have been defined for the user acceptance test phase:

- Test results or reports have been submitted for acceptance test plan and they are verified at QA manager level and by project manager of Concordia University.
- There are no functional or non-functional defects pending in the MiniSIS that may hamper the system to go to next phase that may be production or addition of any new functionality.
- The defects that were found during acceptance test plan have been reworked by development team and are fixed and them QA team has again executed test cases and checked if rework has affected the system positively or not.

5 INFRASTRUCTURE

5.1 Test environments

5.1.1 Acceptance tests

Necessary test environment(s): Below is the list of necessary hardware, software test environments.

- The client must have installed the Java version 5 or 6.
- The transaction database at the University end must be available for the software features to work properly.

5.2 Office setup.

Components	**Comment**
Hardware	
1PC with Windows XP or higher	To test the MiniSIS functionalities
1 Backup system	In case of failure.
Workspace	
Computer Lab	To ensure the testing without any external interface.
Small Office	For any project level meeting.
Built-In Packages	
Microsoft office suite	For creation of test report and other documents.
Others	
Refreshment	For motivating the testers

6 MANAGEMENT

6.1 Test management

All the test activities are governed and supervised by the Quality Assurance team. All the updates regarding the testing activities will be communicated by sending e-mail to project manager and or test manager of client. This test report will contain all the outputs of testing activities that are carried out on the system i.e. all the observed outputs and quality of system. This would provide an idea to the stakeholders to decide next phase of the project.

6.2 Defect procedure

As per the standards a Defect Log is maintained by the Quality Assurance team where as soon as the defect is found it is logged into a word file with the details such as Defect type, its location, Name of the tester and its priority. After this the log is emailed to the developer team for the correction and the defect is tested again for the conformance.

7 ESTIMATION & PLANNING

The planning and estimation for the acceptance test plan must correspond with the master test plan. All deviations must be tuned and approved by the client and a justification for the deviations has to be given, in terms of Result, Risk, Time and Money.

The extent of detail as shown below is the minimal extent of detail at the level of test plan. Where it is needed and possible, apply a higher level of detail.

7.1 Estimation

The estimation in hours is as follows:

Who	**Plan**	**Specification**	**Execution**	**Conclusion**	**Totals**
Test coordinator	6	12	0	8	26
Test specialists/Quality Assurance Team	6	12	8	8	34
Totals:	**12**	**24**	**8**	**16**	**60**

Test plan estimation should not take no more than three to four hours to meet with QA team and discuss the plan rest of time is required for creating logistics, infrastructure and software needs for acceptance testing phase.
Specification time is calculated considering time lag that may be caused in sending and receiving replies to queries and making corrections in the various versions according to replies received in test cases and test plans.
Execution time is calculated on the bases that it will take no more than ten to twelve minutes for executing test cases and for first version of this system will have no more than thirty test cases which accounts to be five to six hours extra time is given in order to comfortably execute all test cases and in some cases test may have to be run more than once in order to confirm outputs.
Towards the end test report analysis and conclusion should take no more than eight hours of which two to three hours are reserved for QA team to meet with client and discus all progress in executing test cases and all the observations that are recorded. This report is signed by Concordia University’s project manager.

7.2 Planning

The activities to be executed have been taken up in the overview below.

Activity	Executioner	Start date dd/mm/yyyy	End date dd/mm/yyyy	Duration	Relations
Planning	QA manager and team	20/10/2017	20/10/2017	6 hours	All the initial testing phases namely unit, unit integration and system testing is already completed and acceptance test plan is made based on system requirements specification (SRS)
Specification	QA manager and team	21/10/2017	21/10/2017	12 hours	When acceptance test plan ready and agreed upon then next step is specification and test cases these are made by entire team and their manager.
Execution	QA team			To be done in the later stages of the project	When acceptance test plan is ready and all specifications and test cases are written and all other non-functional requirements in SRS are also known, then this phase starts

Conclusion	QA team			To be done in the later stages of the project	This phase starts after completion of all the phases and based on outputs of execution phase, a report is created which contains the summary of all the results based upon this report a risk analysis is done then all these documents are shown to stakeholders who decides what is to be done with the project

8 IMPLEMENTATION RESULT ANALYSIS

<table>
<tr><td>Tested by</td><td>QA Team</td></tr>
<tr><td>Related Use Case ID</td><td>UC001</td></tr>
<tr><td>Test Type</td><td>Acceptance Testing</td></tr>
<tr><td>Test Case Number</td><td>TC001</td></tr>
<tr><td>Test Case Description</td><td>To verify whether user has been created</td></tr>
<tr><td colspan="2">Item(s) to be tested</td></tr>
<tr><td>1.</td><td>Needs to be verified that Who has privilege to create user</td></tr>
<tr><td>2.</td><td>e-mail authentication verification needs to be tested</td></tr>
<tr><td>3.</td><td>Password Verification and authentication need to be tested</td></tr>
<tr><td>4.</td><td>Need to be verified that user receives the confirmation e-mail</td></tr>
<tr><td colspan="2">Specification</td></tr>
<tr><td>Input</td><td>Expected Output/Result</td></tr>
<tr><td>1. Availability of feature,” Create User”</td><td>It should be enable only for admin user</td></tr>
<tr><td>2. Verification of E-mail which will be passed in the email parameter.</td><td>Authentication alert should get displayed that email is validated.</td></tr>
<tr><td>3. Password verification</td><td>Password field should be validated as per password’s policy</td></tr>
<tr><td>4. On submitting event, verification of sending the confirmation email to the user.</td><td>Alert should get displayed when the confirmation mail to activate the user has been sent on submitting the create user’s form</td></tr>
<tr><td colspan="2">Procedural Steps</td></tr>
<tr><td>1.1</td><td>If user is admin go to step 1.2. otherwise for other users go to step 1.4.</td></tr>
<tr><td>1.2</td><td>Go the feature,” Create User”</td></tr>
<tr><td>1.3</td><td>This feature should be functional for admin user</td></tr>
<tr><td>1.4</td><td>feature,” Create User” will not be visible</td></tr>
<tr><td>2.1</td><td>Admin enter the email id in the email field</td></tr>
<tr><td>2.2</td><td>System verifies whether email id is validated or not</td></tr>
<tr><td>2.3</td><td>If it is validated, then alert should get populated that email id is authenticated.</td></tr>
<tr><td>3.1</td><td>user enter in password field</td></tr>
<tr><td>3.2</td><td>System verifies the password as per the defined policy for password.</td></tr>
<tr><td>3.3</td><td>Systems pops up the message according i.e. if password is not valid then alert would be ,”incorrect password format”.</td></tr>
<tr><td>4.1</td><td>Admin fills the correct information.</td></tr>
<tr><td>4.2</td><td>Admin submit the information.</td></tr>
</table>

4.3	Admin gets the message that link has been sent to user’s mail.
Implementation Analysis	As per the input specification these are the output. 1. Pass 2. Pass 3. Fail : password conditions are not properly incorporated 4. Fail : I couldn’t add user due to password issue but getting email confirmation.

Tested by	QA Team
Related Use Case ID	UC002
Test Type	Acceptance Testing
Test Case Number	TC002
Test Case Description	To verify User’s account has been activated
Item(s) to be tested	
1.	Needs to be verified that user receives mail or not
2.	Needs to be verified that user gets validated mail
3.	Needs to be verified the hyperlink is valid
4.	Needs to be tested that the user is activated
Specification	
Input	Expected Output/Result
1. Verification of confirmation mail in user’s mail account.	User should receive the confirmation e-mail after creating the user by admin
2. Verification of the content of mail information	The email should have valid content to activate the account.
3. Verification of clicking on the hyper link	Then valid message should be populated while clicking on links
4. Login event validation when user logs in.	User would be able to login after activating his/her account
Procedural Steps	
1.1	User opens his/her respective mail account.
1.2	User searches the confirmation mail
1.3	he/ she receives the mail, if it has been sent to that email id by admin user.
2.1	User open the mail
2.2	Users gets the procedure to activate the account i.e. hyperlinks and other necessary detail.
3.1	Users click on link
3.2	The link populates to activate the account.
3.3	User get the confirmation message that account has been activated.

4.1	User opens the login page
4.2	User enter the credential, if user opens before activating his/her account ,go to step 4.3 ,otherwise step 4.5
4.3	User logs in the software successfully
4.4	User gets an alert to activate the account.
Implementation Analysis	As per the input specification these are the output. 1. Pass 2. Pass 3. Pass 4. Fail : Can not log in due to password authentication issue.

Tested by	QA Team
Related Use Case ID	UC003
Test Type	Acceptance Testing
Test Case Number	TC003
Test Case Description	To verify whether password has been reset or not
Item(s) to be tested	
1.	Hyper link of forget password needs to be worked
2.1	Then information needs to be validated what user enters in the redirected form
2.2	Reset password mail needs to be validated that email has been sent to registered mail
3.1	It needs to be verified that user has received the email.
3.4	Reset password form's hyperlink needs to be verified
4	It needs to be verified that password has been changed successfully.
Specification	
Input	Expected Output/Result
1. Click on forget password hyperlink	Link should be redirected on change password form.
2. Enter information which are required to reset the form i.e. username, email etc. and click on submit button	If information is not matched with authenticated user , then system should get displayed error.
3. Click on received hyperlink of resetting the password in user's email	Password should be reset by receiving the hyperlink of setting the new password.
4. Click on login button after resetting the password.	Access with new set password has been successfully
Procedural Steps	
1.1	User click on forget password hyperlink
1.2	System redirects the link to change password form.

2.1	If user enters the valid information, go to step 2.2 otherwise go to step 2.3
2.2	A message gets displayed that to reset the password, the link has been sent to the registered email
2.3	User gets an alert that provided information is invalid.
3.1	User goes to his registered email, and clicks on received hyperlink
3.2	System redirects the link to page of setting new password.
3.3	User set the new password after set the password as per policies if it is not as per policies ,go to step 3.5
3.4	User gets the message that password has been reset successfully
3.5	User gets an alert that password is wrong.
4.1	User opens the login page
4.2	User enter the credential, if user enters with new changed password, go to step 4.3 ,otherwise step 4.5
4.3	User logs in the software successfully
4.4	User gets an alert that invalid password.
Implementation Analysis	As per the input specification these are the output. 1. Fail: No forget password is redirected. 2. Fail : Can not check 3. Fail : Feature is not available 4. Fail : Feature is not available

Tested by	QA Team
Related Use Case ID	UC004
Test Type	Acceptance Testing
Test Case Number	TC004
Test Case Description	To verify whether phone is verified or not
Item(s) to be tested	
1.	Profile page is working that needs to be verified.
2.	Phone no. format needs to be verified
3.	It needs to be validated that the message of unique code has been sent to entered mobile no.
4.1	The unique code has been matched that needs to be verified.
4.2	It needs to be verified that the mobile no. has been updated/verified
Specification	
Input	Expected Output/Result
1. Click on profile page	System should get displayed the profile page
2. Enter the mobile no.	The mobile no. is not in well format or exceed the length then alert would be displayed accor-

		ding.
3. Visibility of unique code field		The confirmation message would be displayed that the unique code has sent to enter mobile and display the field to enter that code.
4. Enter the unique code.		A message would be popped up that the code has been verified and mobile has been saved
Procedural Steps		
1.1	User enter the credentials	
1.2	User goes to profile page.	
1.3	User clicks on profile page link.	
1.4	System gets the displayed profile form	
2.1	If user is new, go to step 2.2 otherwise go to step 5.1	
2.2	User enter the mobile no.	
2.3	System checks the format and length of mobile no. and displayed the message accordingly.	
2.4	after valid mobile no., system should get displayed the field where user has to enter the valid code.	
3.1	An alert should get displayed that the unique code has been sent to the mobile no.	
4.1	User gets the unique code on mobile.	
4.2	User enters the unique code	
4.3	User should get displayed an alert after validating the unique code whether it is valid or invalid.	
4.4	User gets the message that the mobile no. has been validated.	
5.1	Click on update mobile number and go to the step 2.2 to 4.4	
Implementation Analysis	Fail: As feature is not available	

Tested by	QA Team
Related Use Case ID	UC005
Test Type	Acceptance Testing
Test Case Number	TC005
Test Case Description	To verify whether the user can login or not
Item(s) to be tested	
1.	Login Page needs to be tested whether it is functional or not
2.	Username and password need to be authenticated or not
3.	It needs to be validated that user role is as per the username or not
4	Two factor authentication needs to be verified

Specification	
Input	Expected Output/Result
1. Click on main page	System should get displayed the login page if user is not logged in already
2. Click on login button	The system give access to portal if the credentials are validated.
3. Click on role hyperlink	The role should be validated as per the user who logs in
4.1.Enter email to get code of access 4.2.Enter phone no. to get access	The system must give access only after passing two factor access authentications for new users.
Procedural Steps	
1.1	User opens browser
1.2	User enters the main page
1.3	User gets the login page
2.1	User enters the username and password
2.2	User clicks on login button
2.3	If username and password are matched then login is successful otherwise it will prompt error,” invalid username/password” { if user is new or didn’t change the authentication factor setting go to step 4.1 otherwise step 3.1}
3.1	After login in the portal, user clicks on roles
3.2	If role is related to user name ,go to step 3.3 otherwise step 3.4
3.3	The role is validated successfully.
3.4	The integration between user and role is not correct logically.
4.1	User gets the radio buttons of email /mobile for receiving code to access his/her account
4.2	User gets the unique code on mobile/registered email.
4.2	User enters the unique code in the code field
4.3	User should get displayed an alert after validating the unique code whether it is valid or invalid.
4.4	User accesses the portal if code is validated
Implementation Analysis	As per the input specification these are the output. 1.Pass: user can log in 2. Pass: Page is being redirected 3. Fail : Feature is not available 4. Fail : Feature is not available

Tested by	QA Team

Related Use Case ID	UC006
Test Type	Acceptance Testing
Test Case Number	TC006
Test Case Description	To verify whether the user information is update or not
Item(s) to be tested	
1.	It needs to be checked edit information feature is working or not
2.	It needs to be verified that necessary information is only in edit mode
3.	It needs to be verified that information has been edited successfully.
Specification	
Input	Expected Output/Result
1. Click on edit information button	Fields should be in edit mode
2. Edit parameters verification	Necessary parameter needs to be edited only.
3. Save button event verification	All edited information should be updated successfully.
Procedural Steps	
1.1	TC005 should be passed
1.2	Click on profile page
1.3	Click on edit information button
1.4	Information must be in edit mode i.e edit button is functional.
2.1	Only necessary information must be edited i.e. password, address, phone no., email id and etc.
3.1	After edited all information with in define policies ,click on save buttun
3.2	If save button is functional and information is saved , go to step 3.3 otherwise step 3.4
3.3	The information has been updated successfully.
3.4	The alert should be prompted if any field if any field is edited again the defined validation of respective parameters.
Implementation Analysis	As per the input specification these are the output 1. Fail : no information is available 2. Fail : To change the password or email in account 3. Fail : we can not test as other Info is not available.

Tested by	QA Team
Related Use Case ID	UC007
Test Type	Acceptance Testing
Test Case Number	TC007
Test Case Description	To verify that course has been registered
Item(s) to be tested	
1.	It needs to be tested that courses page is functional

2.	It needs to be verified that course has been selected or not.
3.1	Enroll validation needs to be verified as per the criteria of program, due amount, pre-requisite course.
3.2	It needs to be verified that course has been enrolled.
Specification	
Input	Expected Output/Result
1. Click on course page feature	All courses should get displayed.
2. Select the preferred courses	The preferred course should be selected.
3. Enroll the selected course	The course should be enrolled when all requirements meet to enroll the course.
Procedural Steps	
1.1	TC005 should be passed
1.2	Click on courses page
1.3	If courses are offered, then go to the step 1.4 otherwise step 1.5
1.4	The offered courses are visible for the term (semester).
1.5	System provides an alert that no courses are offered yet for the respective term.
2.1	Select the course
2.2	Course is highlight different from others that means course has been selected
3.1	Click on add validation
3.3	If Course has been enrolled, go to step 3.4 otherwise got to step 3.5
3.4	The course has been enrolled as per respective term successfully
3.5	The course can't be registered in user's program and alert would be prompted as per the following concerned reasons: • Due amount is pending • Pre-requisite course needs to be taken (conditional with completion target date) • Schedule confliction issue i.e. clash the timing of schedule or course is already registered • Selected course is from different department (can be added only by department advisor(conditional)
3.6	If selected course is from different program, then go to step 3.7 otherwise go to step 3.8.
3.7	The message gets displayed that selected course is not from department please contact with own department advisor.
3.8	advisor has privileged to add that course as per student id with in defined program policies.
Implementation Analysis	As per the input specification these are the output 1. Pass: The courses are available to register 2. Pass: course has been selected to register 3. Partially passed : No validation is incorporated at the time to enroll the course.

Tested by	QA Team
Related Use Case ID	UC008
Test Type	Acceptance Testing
Test Case Number	TC008
Test Case Description	To verify that course has been dropped

Item(s) to be tested	
1.	It needs to be tested that courses page is functional
2.	It needs to be verified that course has been selected or not.
3.1	It needs to be verified that when user drop the course it must be validated that user have enrolled that course and the date of availability for that course exists.
3.2	It needs to be verified that course has been dropped from that term.

Specification	
Input	Expected Output/Result
1. Click on course page feature	All courses should get displayed.
2. Select the preferred courses	The preferred course should be selected.
3. Enroll the selected course	The course should be enrolled when all requirements meet to enroll the course.

Procedural Steps	
1.1	TC005 should be passed
1.2	Click on courses page
1.3	If courses are offered, then go to the step 1.4 otherwise step 1.5
1.4	The offered courses are visible for the term (semester).
1.5	System provides an alert that no courses are offered yet for the respective term.
2.1	If user has already enrolled courses in his/her term , go to step 2.2
2.2	If user selects the enrolled course , got step 2.3 otherwise go to step
2.3	User clicks on drop button
2.4	If date is available to drop, then go to step 2.5 otherwise step
2.5	System is prompted a message that the selected course has been dropped
2.6	The status of that course will be changed from enroll which shows that that course is no more
2.7	An alert get displayed that selected course can’t dropped because the deadline is over
2.8	System gets displayed an alert that the select only the enrolled course.

Implementation Analysis	As per the input specification these are the output 1. Pass: The courses are available to register 2. Pass: course has been selected to register 3. Partially passed : No validation is incorporated at the time to enroll the

	course.

Tested by	QA Team
Related Use Case ID	UC009
Test Type	Acceptance Testing
Test Case Number	TC009
Test Case Description	To verify that course has been added in the program for particular term
Item(s) to be tested	
1.	It needs to be verified that manage course is operated by faculty member
2.1	It needs to be verified that course must be unique for each term
2.2	It needs to be verified that the schedule of class must not be clashed of two classes for same instructor during the term
2.3	It needs to be verified that the new course has been added for specific
Specification	
Input	Expected Output/Result
1. Click on course page feature	All courses should get displayed.
2. Availability of add new course	It should be enable for faculty member only
3. Click on add new course	New page should be redirected to add new course
4. Click save new course	New course should be added successfully.
Procedural Steps	
1.1	TC005 should be passed and check the role if it is not faculty member, go to step 1.2 otherwise go to step 2.2
1.2	Click on courses page
2.1	Check add new course feature must not be available
2.2	Click on course page
2.3	Check add new course feature is available
3.1	Click on add new course
3.2	Page is redirected to add new course page
3.3	Enter all the detail about new course as per requirement
4.	Click on save button
4.1	If course is not added before for same term and the schedule is not matched for the same instructor for same term, go to step 4.2 otherwise go to the step 4.3
4.2	System prompts the message that new course has been added success fully for selected term
4.3	An alert gets populated accordingly to stop adding the new course.

2.8	System gets displayed an alert that the select only the enrolled course.
Implementation Analysis	As per the input specification these are the outputs Fail: Feature is not available

Tested by	QA Team
Related Use Case ID	UC010
Test Type	Acceptance Testing
Test Case Number	TC010
Test Case Description	To verify that course has been registered by faculty member

Item(s) to be tested	
1.	It needs to be tested that courses page is functional
2.	It needs to be verified that course has been selected or not.
3.1	Availability Student search field need to be verified for advisor of concerned department
3.2	Enroll validation needs to be verified as per the criteria of program, due amount, prerequisite course.
3.3	It needs to be verified that course has been enrolled.

Specification	
Input	Expected Output/Result
1.1. Click on course page feature	All courses should get displayed.
1.2. Availability of search student	It should be enable only for advisor
2. Select the preferred courses	The preferred course should be selected.
3. Enroll the selected course	The course should be enrolled when all requirements meet to enroll the course.

Procedural Steps	
1.1	TC005 should be passed only for advisor
1.2	Click on courses page
1.3	Check search student , it must be available only for advisor of department
1.3	If courses are offered, then go to the step 1.4 otherwise step 1.5
1.4	The offered courses are visible for the term (semester).
1.5	System provides an alert that no courses are offered yet for the respective term.
2.1	Select the course
2.2	Course is highlight different from others that means course has been selected
3.1	Click on add validation
3.3	If Course has been enrolled, go to step 3.4 otherwise got to step 3.5

3.4	The course has been enrolled as per respective term successfully
3.5	The course can’t be registered in user’s program and alert would be prompted as per the following concerned reasons: • Due amount is pending • Pre-requisite course needs to be taken (conditional with completion target date) • Schedule confliction issue i.e. clash the timing of schedule or course is already registered • Selected course is from different department (can be added only by department advisor(conditional)
3.6	If selected course is from different program or other issue, then go to step 3.7 otherwise go to step 3.8.
3.7	The message gets displayed that selected course is not from department please contact with own department advisor.
3.8	advisor has privileged to add that course as per student id with in defined program policies.
3.9	The course is added successfully as per term and condition for enrolling the courses.
Implementation Analysis	Fail: feature is not available.

Tested by	QA Team
Related Use Case ID	UC011
Test Type	Acceptance Testing
Test Case Number	TC011
Test Case Description	To verify that transcript can be viewed by advisor or student
Item(s) to be tested	
1.	It needs to be tested that login has been successful
2.	It needs to be verified that report page is active for both users
3.1	It needs to be verified that when user is advisor the search student id should be available
3.2	It needs to be verified that the transcript is viewed
Specification	
Input	Expected Output/Result
1. Click on report page	Report page should get displayed.
2. Search student ID feature availability	It should be available for advisor
3. Click on view transcript	Transcript should be viewed
Procedural Steps	
1.1	TC005 should be passed
1.2	Click on report page
1.3	It should be redirect on transcript page
2.1	If user is advisor, got to step 2.2 otherwise go to step 3.1 for other users

2.2	The search student id is enable
2.3	Search the student id
2.4	Gird is available for search id
3.1	Click on view transcript button
3.2	If student has completed any course or term, go to step 3.3 otherwise go to step 3.4
3.3	The transcript will be generated as per courses
3.4	The blank template will be generated
Implementation Analysis	Fail: feature is not available.

Tested by	QA Team
Related Use Case ID	UC012
Test Type	Acceptance Testing
Test Case Number	TC012
Test Case Descripti-on	To verify that tuition fees has been paid complete or not

Item(s) to be tested	
1.	It needs to be tested that tuition fee need to be paid or not
2.	It needs to be verified that tuition fee is pending or not
3.1	It needs to be tested that the amount has been payed successfully
3.2	It needs to be verified that pending amount is being reduced after paying the fees or not
4	It needs to be verified the penalty is applied or not as per due date

Specification	
Input	Expected Output/Result
1. Click on payment page	Payment page should get displayed
2.1. Click on pay button	Payment information should be redirected
2.2. Enter Amount	The float/integer value should be acceptable
2.3. Select payment method	Payment method should be selected

2.4. Click on submit	Payment has done successfully
3. Due date	Penalty should be applied after due date.
Procedural Steps	
1.1	TC005 should be passed
1.2	Click on payment page
1.3	It should be redirect on payment page
2.1	If Payment is pending, go to step 2.2 otherwise step 2.4
2.2	Due amount will have more than 0 value, it shows that much balance need to be paid
2.3	System gets displayed no amount due
2.4	If amount is pending , click on pay button
2.5	Payment information page needs to be redirected
2.6	Enter the amount
2.7	If value in amount is string or alphanumerical , it gets displayed alert.
2.9	If value is integer or float , the value is being passed in amount field
3	Select the payment method
4.	The field is visible according to selection method of payment
5	Click on submit
6	If information is correct, go to step 7 otherwise go to step 9
7	The system gets displayed a message that the entered amount has been paid successfully
8	The due is reduced accordingly
9.	The alerts get displayed that the information is incorrect
10	If Due amount is pending after due date, go to step 10.1 otherwise go to step 10.3
10.1	The penalty is implement as late fee
10.2	The due amount displays accordingly
10.3	Due amount remains same till due date
Implementation Analysis	As per the input specification these are the outputs 1. Pass: Payment page is redirected. 2. Fail: Every time it shows invalid card no. 3. Pending: Due to error shown in the paper.

Tested by	QA Team
Related Use Case ID	UC013
Test Type	Acceptance Testing
Test Case Number	TC013
Test Case Description	To verify whether grades are posted or not
Item(s) to be tested	
1.	It needs to be tested that only instructor must have ADD Grades Feature
2.	It needs to be verified that all enrolled students should be visible only.
3	It needs to be verified that grades are being selected for the students
4	It needs to be verified that grades have been posted
Specification	
Input	Expected Output/Result
1.1. Click on course page feature	Course pages should be opened
1.2. Availability of ADD Course	It should be enable only for instructors
2. Click on ADD Grades	It should be redirected to add grade page
3. Select Grade for students	Grade should be selected with respect of student
4. Click on submit to post the grades	The grades should be posted for all enrolled students
Procedural Steps	
1.1	TC005 should be passed only for advisor/student
1.2	Click on courses page
1.3	Check Add Grades feature, it must be available only for Instructor of department
2.1	Click on ADD Grades
2.2	System redirects on add grades page
2.3	If course has not been taken at least one student, go to 2.4 otherwise go to step 2.5
2.4	System displays, "no student is enrolled yet"
2.5	System displays Enrolled student and their ID with a drop down box to add grades
3	Select grades for each enrolled student
4.1	Click on Submit before selecting grades go to 4.2 otherwise go to 4.3
4.2	If any instruct didn't select grades for any student, alert get displayed accordingly.
4.3	System displays that grades are posted successfully and has been notified to all students
4.4	Student gets notification on mail/phone

5.1	Student open his account
5.2	Student have following option to check grades: 1. View Transcript (follow test case – TC011) 2. Go to enrolled courses.
5.3	If grade is available for specific course, go to step 5.4 otherwise go to step 5.5
5.4	The grade has been posted successfully
5.5	System is not working perfectly
Implementation Analysis	Fail: Feature is not incorporated

Tested by	QA Team
Related NFR	Security & Reliability
Test Type	Acceptance Testing
Test Case Number	TC014
Test Case Description	To verify whether Software is secure or not
Item(s) to be tested	
1.	It needs to be verified that password should be saved in encrypted format
2.	It needs to be verified that password policies are optimized
3	All privileges(rights) for user should be configured accurately i.e. role and responsibility should be appropriates
4.	It needs to be verified that bank detail is encrypted and secure or not.
Specification	
Input	Expected Output/Result
1. Password Field	Password should be saved in encrypted form
2. Availability Roles and responsibility	It should be defined as per users' credentials
3. Enter bank Detail	It should in secure environment
Procedural Steps	
1.	Password encryption validation should be on following places: 1) User registration 2) Change Password 3) Saved password in database 4) Log Page 5) Pay tuition
2.	Above tests cases showed few roles and responsibility which should configured according to secure the system
Implementation Ana-	As per the input specification these are the outputs:

lysis	1. Pass: password is encrypted and log in and logout page is working fine. 2. Fail: role and responsibility is not working as per expected result as for example course should be assign to instructor only by admin but student name is also incorporated.

Tested by	QA Team
Related NFR	Performance
Test Type	Acceptance Testing
Test Case Number	TC015
Test Case Description	To verify whether Session times are applied
Item(s) to be tested	
1.	It needs to be verified that the processing speed is optimized
2.	Proper session is created and maintained for all users
3	Software performance in case of heavy load.
Procedural Steps	
1.	Multiple users can log together
2.	Session time out is maintained if it is not
3	It conflicts the processing i.e. performance is slow
Implementation Analysis	As per the input specification these are the output 1. Pass: as per configured user , it is working fine. 2. Fail: Session is not maintained properly as no internet is available but still when internet connectivity will be started user account will exist i.e no session time out .

APPENDIX 1 – PRODUCT RISK ANALYSIS

Table Test goals

Type of test goal	Examples	Relevant characteristics
Acceptance by client	Add Course Drop Course View Transcript Pay fee Add grades	Reliability Usability Security Portability

Risk table

Nr	Event	Consequence	Impact	Chance	Countermeasures	Owner
1	Test cases fail for success scenario.	Developers should re work on that code as well all the other related block of codes and it will cost time and money.	H	L	Development team should make sure by doing a thorough unit and integration testing	Development team
2	Test cases fail but with appropriate error messages.	There might be some logical or design errors. Developers will need to re-work the code but it is manageable	M	M	Developers should repetitively test the code using different types of testing	Development team

3	Test cases designed did not test all possible scenarios errors went un detected.	It’s a very critical failure because after implementation stakeholders may face problems and this would result in immediate rework or replacement of entire software this can result in major time and financial losses.	H	L	QA team should create test cases very carefully based upon latest documents from requirements and development team. It should be made sure that all the documents are from latest versions. After creating test cases entire QA team should go through the test cases for any incompleteness and document should be formally signed by project manager of Concordia University	QA Team
4	All Test cases are not executed and errors left out.	This is a very dangerous scenario as after implementation stakeholders may face problems and this would result in immediate rework or replac ment of entire software this can result in major time and financial losses	H	L	QA manager should take feedback from his team should keep a proper check on testing he should also check word processing documents related to testing.	QA Team

References:

[1] Acceptance Test Plan Template provided by Dr. Rachida Dssouli